

МГТУ им. Н. Э. Баумана, кафедра ИУ5
курс “Разработка интернет-приложений”

Лабораторная работа №2

Python.

Объектно-ориентированные возможности

выполнил: Васильев Александр
студент группы ИУ5-54

Москва 2018

2) Задание

Необходимо разработать программу, реализующую работу с классами.

1. Программа должна быть разработана в виде консольного приложения на языке Python 3.
2. Все файлы проекта (кроме основного файла main.py) должны располагаться в пакете lab_python_oop.
3. Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета lab_python_oop.
 1. Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры. Подробнее про абстрактные классы и методы Вы можете прочитать здесь - <https://docs.python.org/3/library/abc.html>
 2. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать здесь - <https://docs.python.org/3/library/functions.html#property>
 3. Класс «Прямоугольник» наследуется от класса «Геометрическая фигура». Класс должен содержать конструктор по параметрам «ширина», «высота» и «цвет». В конструкторе создается объект класса «Цвет фигуры» для хранения цвета. Класс должен переопределять метод, вычисляющий площадь фигуры.
 4. Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус». Для вычисления площади используется константа math.pi из модуля math - <https://docs.python.org/3/library/math.html>
 5. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны.
 6. Для классов «Прямоугольник», «Квадрат», «Круг»:
 1. Определите метод "repr", который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод format - <https://pyformat.info/>
 2. Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
4. В корневом каталоге проекта создайте файл main.py для тестирования Ваших классов (используйте следующую конструкцию - <https://docs.python.org/3/library/main.html>).
Создайте следующие объекты и выведите о них информацию в консоль:
 1. Прямоугольник синего цвета шириной 3 и высотой 2.
 2. Круг зеленого цвета радиусом 5.
 3. Квадрат красного цвета со стороной 5.

3) Исходный код

GeometricFigure.py

```
from abc import ABC, abstractmethod

class GeometricFigure(ABC):
    @abstractmethod
    def area(self):
        pass
```

FigureColor.py

```
class FigureColor:
    def __init__(self, col):
        self._color = col
    def color(self):
        return self._color
```

Rectangle.py

```
from lab_python_oop.GeometricFigure import GeometricFigure as GeomF
from lab_python_oop.FigureColor import FigureColor as FigCol
```

```
class Rectangle(GeomF):
    def __init__(self, s, w, l):
        col = FigCol(s)
        self.width = w
        self.length = l
        self.color = col.color()
        self.name = 'Rectangle'
    def getname(self):
        return self.name
    def area(self):
        return self.length * self.width
    def __repr__(self):
        return "type: " + str(self.getname()) + " width = " + str(self.width) + "
length = " + str(self.length) + " area = " + str(self.area()) + " color: " +
str(self.color)
```

Square.py

```
from lab_python_oop.Rectangle import Rectangle
class Square(Rectangle):
    def __init__(self, s, w):
        self.len = w
        super().__init__(s, w, w)
        self.name = 'Square'
    def getname(self):
        return self.name
    def area(self):
        return self.len * self.len
    def __repr__(self):
        return "type: " + str(self.getname()) + " length = " + str(self.len) + "
area = " + str(self.area()) + " color: " + str(self.color)
```

Circle.py

```
from lab_python_oop.GeometricFigure import GeometricFigure
from lab_python_oop.FigureColor import FigureColor
from math import pi
class Circle(GeometricFigure):
    def __init__(self,s,r):
        self.rad = r
        col = FigureColor(s)
        self.color = col.color()
        self.name = 'Circle'
    def getname(self):
        return self.name
    def area(self):
        return self.rad * self.rad * pi
    def __repr__(self):
        return "type: " + str(self.getname()) + " radius = " + str(self.rad) + "
area = " + str(self.area()) + " color: " + str(self.color)
```

main.py

```
from lab_python_oop.Rectangle import Rectangle
from lab_python_oop.Square import Square
from lab_python_oop.Circle import Circle
rectangle = Rectangle('blue', 3, 2)
square = Square('red',5)
circle = Circle('green',5)
print(rectangle.__repr__())
print(square.__repr__())
print(circle.__repr__())
```

4) Скриншоты с результатами выполнения

